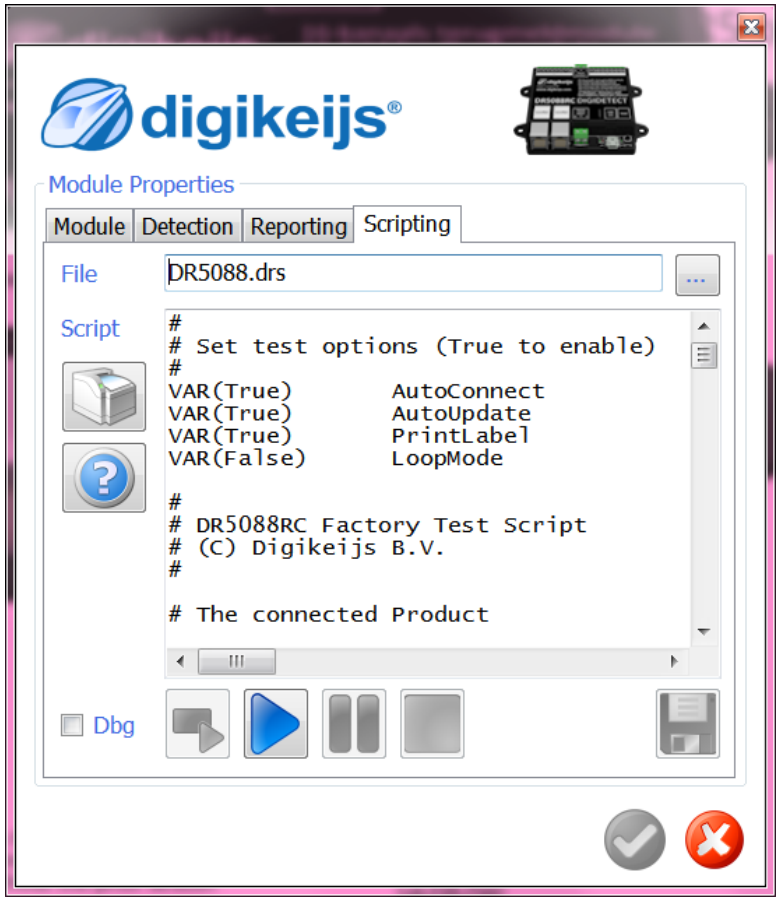
Dr.Script

A Scripting Language to contol your DR5xxx products

Preliminairy **Version** *0.9 (minimum Firmware 1.4.8)*



1 Intro

Already since the introduction of our DR5000, all the the products in the DR5xxx line contain the Dr.Command command language/protocol.
To make easy use of Dr.Command and it's power, we developed Dr.Script.
Dr.Script is a BASIC/Assembly like scripting language, with which you can send an receive Dr.Command messages as well as access many internal features directly from your DR5xxx product.
This manual does not pretend to be complete… it merely lists the basic syntax and explains the Build-in statement elements.

2 Syntax

In general Dr.Scripts recognizes three different type of script lines: 1) Comment lines, 2) Declaration lines, 3) Assignment lines, 4) executable lines and 5) Label Lines

2.1 Comment lines

Comments in Dr.Script start with a # character before any other non-white-space character and last for the whole line.
characters elsewhere in a line are taken literally.

2.2 Declaration lines

To declare variables for use in the script, a declaration line consist of the keyword VAR followed by the variable name. Optionally the VARiable can be given an initial value by providing the value in between ( ) e.g. VAR**(False)** UseThisVariable

2.3 Assignment lines

To assign values to pre-declared variables (VAR), an assignment line consists of:

<left-hand> = <right-hand>

Where <left-hand> is the name (case sensitive) of a previously defined variable
Where ' = ' is the assignment operator, surrounded with white-space (at least 1 space or tab before and at least 1 space or tab behind the '=')
Where <right-hand> is an expression of any literal, variable content or built-in function result.
<right hand> may contain the operators : '+', '-', '*', '/'. For addition, subtraction, multiplication and division. In case these operators exist, Dr.Scripts splits the expression from left to right and evaluates the split parts from right to left.

2.4 Executable lines

To execute built-in functions and or flow control statements, Executable Lines take the form of:

<keyword>[(<parms>)] <args>

Where (<parms>) is optional., as well as <args>. <args> may contain results of built-in function execution as last element on the line.

Built in functions always return a string value. Some of which can be converted to numbers or Boolean (False/True).

2.5 Label lines

Labels in Dr.Script occur on as one whole word followed by a ':' character. The line Should not contain other content. Labels are used as targets for flow-control statements like GOTO, JUMP, GOSUB and CALL.

Format:

<label>:

2.6 Variable content and Execution result

To insert the current content of a variable or the result of a built-in function, this value can be obtained by preceding the keyword or variable name by the $ character.

Example:

```
VAR      Version                  // defines the variable 'Version'.
Version  = $VERSION               // gets the DR5xxx products FW version and assigns it to 'Version'.
LOG      $Version                 // Logs the value of 'Version' to the logging window.
```

2 Built-in functions and Keywords

2.1 Keywords

Dr.Script contains some very basic keywords.

Keyword	parms	args	return	Description
VAR	<optional> Initial value	Variable name	-	Declare and optionally initialize a variable
STOP | EXIT	-	-	-	Terminates the execution of the script
PAUSE	<optional> Time to pause in milliseconds	<optional> Time to pause in milliseconds	-	Suspends the execution of the script. A timeout value can optionally be supplied in either <parms> or <args>
GOTO | JUMP	-	Pre-declared labelname	-	Continue script-execution at the specified label
GOSUB | CALL	-	Pre-declared labelname	-	Continue script-execution at the specified label and return to the next statement afte GOSUB | CALL when a RETURN is encountered
RETURN	-	-	-	RETURN to the next line after the calling GOSUB | CALL statement

2.1 Keywords (2)

Keyword	**parms**	**args**	**return**	**Description**
IF	Condition expression	Executable line		<parms> contains a condition expression which consist of a <left> and <right> part. Both parts being separated by one of these compare operators: '=' , '==' , '!=' , '<>' , '>' , '<', '>=' , '<=' <args> contains a regular Executable line. When the expression renders 'True', <args> is executed.

2.1 Built-in functions

2.1.1 Communication functions

Keyword	parms	args	return	Description
FLUSH	-	-	-	Empty the Dr.Scripts communication input queue.
SEND	-	A Dr.Command message. “DR.” is put by SEND and must NOT be specified.	True | False depending on success.	Send a Dr.Command message directly to the DR5xxx product.
WAITFOR	<optional> Timout in milliseconds	Reglar expression for the script to receive from the DR5xxx product.	False in case of timeout, otherwise the matched text	Wait for the texst in <args> to be received from the DR5xxx product. The text may be literal or in the form of a Microsoft.NET regular expression

2.1 Built-in functions(2)

2.1.2 Info functions

Keyword	parms	args	return	Description
VERSION	-	-	Actual Firmware version of the connected product	
APPVERSION	-	-	Actual version of the DR5xxx Configuration App	
PRODUCT	-	-	The name of the connected DR5xxx product	
SERIAL	-	-	Actual serial number of the connected product	

2.1 Built-in functions(3)

2.1.2 Control functions

Keyword	parms	args	return	Description
POWER	<optional> timeout in milli-seconds	On | Off | ?	True | False | <actual state>	Sets (On | Off) or reads (?) the power state
TURNOUT	<optional> timeout in milli-seconds	<address> <state | ? > state: '|' = closed '/' = thrown	True | False | <actual state>	Sets or reads (?) the turn-out state
SPEED	<optional> timeout in milli-seconds	<address> <speed | ? >	True | False | <actual speed>	Sets or reads (?) the loco speed
DIRECTION	<optional> timeout in milli-seconds	<address> <direction | ? > direction: '>' = forward '<' = backward	True | False | <actual direc-tion>	Sets or reads (?) the loco direction
FUNCTION	<optional> timeout in milli-seconds	<address> <functions| ? > functions: function pattern as in Dr.Command (See Log Window)	True | False | <actual func-tions>	Sets or reads (?) the loco functions